古典和歌からの知識発見
－モビルスーツを着た国文学者－

竹田 正幸
九州大学大学院システム情報科学研究院／
科学技術振興事業団さきがけ研究21
takeda@i.kyushu-u.ac.jp

福田 智子
純真女子短期大学
tomoko-f@muc.biglobe.ne.jp

◎文学作品からの知識発見

平成10年より3年間，文部科学省科学研究費補助金特定領域研究（A）「巨大学術社会情報からの知識発見に関する基礎研究」（略称：発見科学）が，九州大学有川節夫教授を領域代表者とし，約60人の研究者を擁した一大プロジェクトとして遂行された．プロジェクトは5つの研究班から構成され，筆者らは第4班「巨大データベースからの知識発見に関する研究」に参加した．この班では「地球物理学から文学まで」を謳い文句に，物理学，化学，生物学，経済等，さまざまな分野の研究者が，情報科学者と連携して，それぞれの視点から研究を展開した．この班の方針は，文献1）にみえる次の言葉に象徴される ― たとえばUCI Machine Learning Repositoryなどの現場から切り離された「よく整理された」データやトイサイズのデータを用いた人工知能研究で，科学等の領域に直接的なそして積極的なかかわりを持っていない研究は，もはや発見科学の研究ではない ―．

では，領域研究者が日頃格闘している「本物の」巨大データから「掛け値なしの」発見を行うためには，何が必要なのだろうか．Fayyadら（1996）は，知識発見のプロセスを解析し，それを「選択・前処理・変換・データマイニング・解釈」という一連のプロセスと捉えている．BrachmanとAnand（1996）は，このプロセスには人間とのインタラクションが自然に現れ，そのインタラクションが重要であることを強調している．Langley（1998）も，計算機を使った発見のプロセスを同じように捉え（図-1），発見システムにどのように人間が介入するかが成功の鍵であり，したがって，発見システムは人間がもっと積極的に発見プロセスに介入できるように支援すべきである，と主張している．

筆者らは「文学作品からの知識発見」というテーマにて，特に古典和歌を扱った研究を展開した．自然界に生起する事象を対象とし，可能な限り主観を排したものと理解されている自然科学分野においてすら，計算機を用いた発見プロセスには人間の介入が不可欠というのである．とすれば，ある意味で主観的であらざるを得ない文学研究においては，人間による介入はいっそう重要なものとなろう．

そこで本研究の成功の鍵は，文学の専門家が主体となり，その力を計算機パワーの上にまっすぐに載せることだと考えられる．いつか文学を理解する計算機プログラムを作りたい，などという夢を追いかけることは，この研究では許されない．もし，古典的な人工知能研究を，人間と同等な自律型ロボットの開発にたとえるならば，**発見科学は，ロボットではなくモビルスーツ☆1を開発する**のだといえるかもしれない．その目的は，装着した人間の力を最大限に引き出すことだ．

研究の主体はあくまで人間とし，計算機は脇役に徹する．そして願わくは「脇役なしでこの発見は有り得な

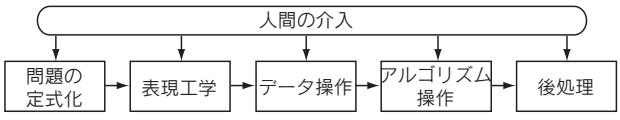

図-1
Langleyによる知識発見のプロセス

☆1 モビルスーツ（MOBILE SUIT）といわれてもピンとこない世代の方々，誠にすみませんが，ロボット風の外観の，戦闘用強化服のようなものをご想像ください．

かった」といわしめたい．筆者らは，このような立場で研究を進めた．なお，第1筆者は情報科学を専門とし，第2筆者は，国文学特に和歌を専門としている．

◎国文学研究における計算機利用

国文学研究における計算機の利用法は，ワープロを別にすると，主として次の2つである．

- エディタの機能やデータベースソフトを用いた索引☆2の作成．
- 全文データを対象としたキーワード検索に基づく用例収集．

これらの利用法は，従来人手で行っていた単純作業から研究者を解放し，より創造的な仕事へ専念させるものと評価できる．だが一方，その気になれば人手でも行える作業を一部分肩代わりしたものともいえるわけで，このことから「単なる時間短縮に過ぎない」という批判も存する．実際，キーワード検索による用例収集を考えてみても，書物の形態をとった「索引」が存在する以上，計算機を使う・使わないによる本質的な差異はない．それでは，文学研究の方法に大きな変化をもたらすような技術は，有り得ないのだろうか．

この点について，村田[6]は，「万葉短歌の字余り法則の再検討」という事例を通じて，計算機利用による「時間短縮」はこれまで考えられていた「時間短縮」とは比較できないほど激しく，従来人手では事実上不可能であった検証作業を可能にすると主張している．村田は，「時間短縮」という一見手順にしか見えないものも，その対象の総体が1人の研究者に与えられた時間の総量をはるかに超えてしまうものであるとき，方法に転化する可能性があるという．

村田の主張は，研究者が主観的な印象としては気がついていながら客観的に検証できないでいた種々の仮説が，計算機の利用により検証可能になるというものだ．実際，伊藤[3]，村上ら[7]，近藤[4]などの先駆的研究はこの線に沿ったものといえよう．だが，筆者らはこれをもう一歩進めて，既存の仮説を検証するのではなく，研究者が新たな仮説を得るプロセスを計算機によって支援することを目指した（図-2）．

ここで，キーワード検索に基づく用例収集は，研究者が任意の表現にまず注目し，次にその用例を収集するといったシナリオを前提としていることに注意されたい．そこでは，研究者がどのような表現に着目するかが，研究の成否を分ける鍵となる．もし，この着目すべき表現について計算機が何らかの指針を与えてくれたとすれば，そこから研究の糸口が得られることが期待できる．すなわち，経験と専門的「勘」に頼るしかなかった研究者のヒラメキを，計算機によって支援しようというのであり，この点が筆者らの研究の大きな特徴である．

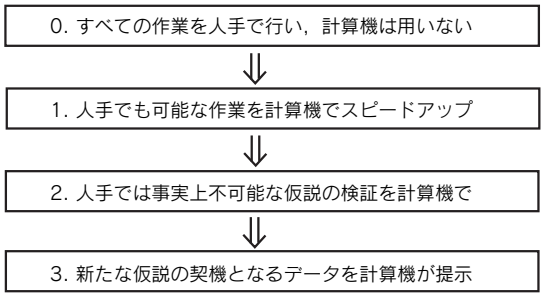

図-2
多くの研究者の利用形態は1であるが，いまだ0を貫く研究者も少なくない．少数の研究者だけが2の形態での計算機利用を試みている．筆者らは3に挑戦した．

国文学研究における計算機の利用

筆者らの研究のもう1つの特徴は，「言い回し」への着目である．古典和歌における表現の類似性の分析は，これまで，名詞や動詞を中心とする自立語に着目して行われてきた．これらの語は，表現の素材となり，「梅に鶯」「紅葉と鹿」のように，特定の組合せで用いられる．しかし，和歌は自立語のみで成り立っているのではない．自立語と自立語を連繋させ一首の和歌にまとめあげるという重要な役割は，付属語（助詞・助動詞）が担う．にもかかわらず，古典和歌の表現研究は歌の素材ばかりを扱い，言い回しをとり上げたものは少ない．

そもそも，付属語は意味を担わないため，それ自体が一首の歌の骨組みを形成していても，研究者の記憶には残りにくい．したがって，研究の糸口が，経験や勘からは，なかなか得られないのである．だが，和歌の類似性を指摘し，表現の影響関係を明らかにするには，伝統的な歌ことばの組合せとともに，言い回しの類似性をも視野に入れる必要がある．

現在，和歌研究者が似た歌を収集しようとするとき，まず使用するのは，『新編国歌大観』（角川書店，1983～1992年）である．この10巻20冊の本は，古典和歌を網羅的に集めた上に，各句索引を備えている．この索引

☆2 ここで「索引」とは，語（あるいは語句）を見出しに立て，見出しごとにその生起位置などの情報を付与したものを指す．この「索引」は，あくまで書物として出版され，利用されるのである．

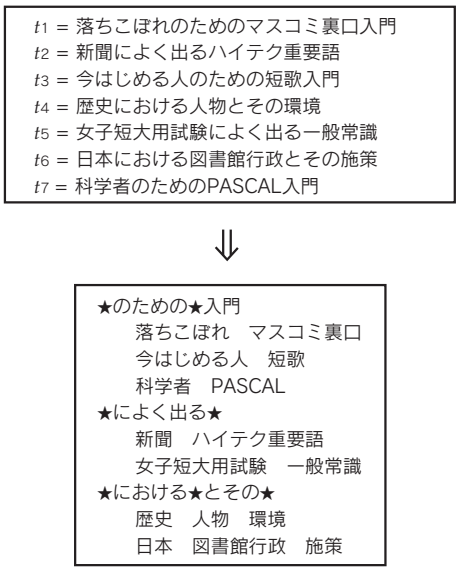

図-3

上に示した$t_1,\ldots, t_7$は，いずれも実在する本の表題である．これらを眺めていると，たとえば，t_1, t_3, t_7にはパターン「★のための★入門」が現れていることに気づく．同様に，t_2, t_5には「★によく出る★」，t_4, t_5には「★における★とその★」が，それぞれ共通に生起している．そこで，それぞれのグループごとに，共通パターンを一度だけ書き，そのあとに，★に合致した文字列を書き並べると，すべての表題を記述できる．このような記述法を，パターンによる符号化と呼ぶ．パターンの選び方，グループの分け方は，もちろん，一意ではない．たとえば，t_1, t_3, t_4, t_7には「★の★」が，t_2, t_5, t_6には「★に★る★」が生起しているので，それに沿った符号化も可能だ．しかし，パターン集合としては，明らかに先に示したものの方が「よい」と思われる．この判断を形式的に行うために，最小記述長原理を用いる．すなわち，記述長を最小にするパターン群を最良とみなすのである．

パターンによる符号化と最小記述長原理

は，句ごとに類似する歌を調べるには便利であるが，句頭の文字から五十音順に配列されているため，句頭にくる自立語におのずと焦点が絞られる．したがって，句頭にこない自立語の検索ができないばかりか，付属語に負うところの大きい言い回しにまで目配りできない．『新編国歌大観』CD-ROM版（角川書店，1996年）が発行され，句末からの検索や本文に対する部分文字列検索が可能になったが，検索すべき表現を着想しなければならない点については，状況は少しも変わっていない．

そこで，筆者らは次の3つに取り組んできた．

1. 特徴パターンの抽出．
2. 類似歌の抽出．
3. 差異表現の抽出．

以下では，その各々について概説する．詳細については，文献12），10），11）をそれぞれ参照されたい．

◎特徴パターンの抽出

和歌の集合から特徴的な言い回しを半自動的に抽出したい．すなわち，計算機が特徴的な言い回しの候補を抽出し，研究者がそれを吟味するのである．言い回しのモデルとして，筆者らは「★せば★ざらましを★」のようなワイルドカード付きパターンを採用した．ここで，ワイルドカード★は，任意の文字列と合致するものとする．そして，先に述べた付属語重視の発想に基づき，パターンの定数部分（★で区切られた部分）を，付属語や用言の活用語尾などの作る文字列に限定した．言い回しを構成するこのような文字列を広い意味で**付属語列**と呼ぶことにしよう．

Σを文字の有限集合とし，可能なパターン全体の集合をΠで表す．SをΣ上の文字列の有限集合とする．パターンの集合$P \subseteq \Pi$がSの**被覆**であるとは，S中の任意の文字列wに対して，wに合致するパターンがP中に少なくとも1つ存在するときをいう．文字列集合Sに対する被覆Pの「良さ」を$good(P, S)$で表すことにすると，我々の問題は次のような最適化問題として定式化できる．

最適被覆問題.
入力：文字列の有限集合$S \subseteq \Sigma^+$.
出力：$good(P, S)$の値を最大化するSの被覆$P \subseteq \Pi$.

問題は，いかにして有効な指標$good$を定めるか，である．筆者らは，この指標を，Brāzmaら（1996）にならい，最小記述長原理に基づいて与えることにした．詳細は省くが，基本的アイディアを図-3に例示したので，ごらんいただきたい．すなわち，集合Sの記述長からSの被覆であるパターン集合Pを用いて符号化した際の記述長を引いたものを$good(P, S)$の値とするのである．その際，上の最適化問題は一般にはNP-困難となる．

筆者らは，いくつかの具体的な歌集に対して，この問題を解く近似アルゴリズムを用いてパターンを抽出し，その結果を評価した．得られたパターンの歌集ごとの相違は，歌人の個性や時代の好みを反映しており，研究者に非常に興味深い視点を与えるものであった．

ここでは，パターンの定数部分を付属語列に限定しており，この「付属語列」として定義された文字列の集合が，領域知識だということになる．この領域知識をシステムに与える際には，助詞，助動詞，および活用語の活用語尾をリストアップし，それらの承接規則を

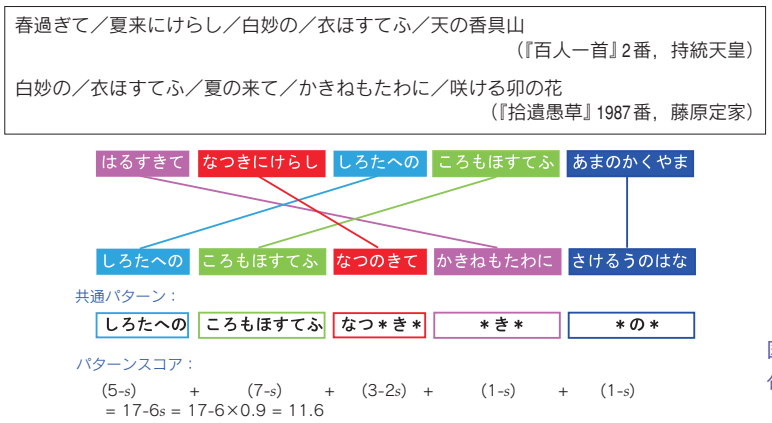

図-4
句の入れ替わりとワイルドカード付きパターンでみる類似性

文脈自由文法として記述した．

和歌をかな文字の連鎖とみなし，あくまで字面での文字列照合を行うわけだから，当然，付属語誤認の問題が生じる．だが，これを何とかしようという第1筆者の気持ちとは逆に，和歌文学者である第2著者の要望は，むしろ，自立語・付属語の区別なく，単なる文字列として表現上の類似性を扱いたい，というものだった．自立語と付属語の区別がそもそも明確なものではない上に，言い回しの一部として自立語が用いられることも少なくない．したがって，あらかじめ言い回しを形成する語の列を限定すれば，想定しなかった未知の言い回しを抽出できない．本来研究の成果として得られるべき知識を，あらかじめ領域知識として与えよ，というのは本末転倒ではないか．

こうして，筆者らは，和歌を単なるかな文字の列とみなし，和歌の類似性を，あくまで文字列間の類似性として扱った研究へと踏み出していくことになった．

◎類似歌の抽出

類似歌の抽出法として，和歌間の類似性指標を定義し，その指標の値の大きい和歌の対を人手により検証する，といった方式が考えられる．このような方式においては，成功の鍵は，類似性指標をいかに定義するかにかかっている．

1つの方法として，類似歌対と非類似歌対のデータを集め，それを訓練例として用いて類似性指標を学習する方法が考えられる．しかし，本研究では，人手による従来研究を計算機になぞらせることではなく，人手では指摘しにくいタイプの類似歌を新たに拾い出すことを目指している．したがって，その意味では，学習に必要な訓練例は得られないものとしなければならない．実際，これまで指摘されてきた類似歌の事例は，専ら自立語を中心としたものであり，言い回しに着目した研究は，あまり行われていない．そこで，類似性指標は，機械学習によらず人手で設計することにした．

文字列の類似性

これまで，文字列間の類似性指標といえば，編集距離（edit distance）が用いられてきた．2つの文字列間の編集距離は，一方を他方に変換するために必要な編集操作の適用回数の最小値として定義される．編集操作として，(1) 文字列中の1文字を別の文字に置き換える操作，(2) 文字列中の1文字を削除する操作，(3) 文字列中の任意の位置に1文字を挿入する操作，の3つを考える．ここでは，それぞれの操作にかかるコストをすべて1と考えているが，より一般的に，編集操作に関与する文字に依存してコストを割り当てるものもある．これを重み付き編集距離（weighted edit distance）と呼ぶ．編集距離は，その名のとおり，文書の編集を行う際のスペルミスの検出や半自動修正などに用いられる．また，DNAの塩基配列やアミノ酸配列を対象とした相同配列検索においても，重み付き編集距離を拡張したものが用いられる．

しかし，編集距離は，元々人間の犯す打鍵ミスをモデル化して作ったものであり，和歌の類似性を扱う際にこれをそのまま適用できるとは考えにくい．そこで，筆者らは，文字列間の類似性指標を設計するための枠組みを新たに導入した．この枠組みでは，2つの文字列間の類似度を，両者に共通するパターンの最大スコアとする．したがって，類似性指標は，(1) パターンの族 Π と，(2) Π 中のパターンに実数値を割り当てる関数，の2つから成る形式的体系である．この体系を，SRS（String Resemblance System）と呼ぶ．この枠組みによれば，「どのような形式のパターンによって文字列間の

かはづなく／ゐでの山吹／ちりにけり／花のさかりに／あはましものを
（『古今集』125番，よみ人しらず）

あしびきの／山吹の花／ちりにけり／ゐでのかはづは／いまやなくらむ
（『新古今集』162番，藤原興風）

かはづなく　ゐでのやまぶき　ちりにけり　はなのさかりに　あはましものを

あしびきの　やまぶきのはな　ちりにけり　ゐでのかはづは　いまやなくらむ

共通パターン：
かはづ　なく　ゐでの　やまぶき　ちりにけり　はな

パターンスコア：
(3) + (2) + (3) + (4) + (5) + (2)
= 19

図-5
断片パターンでみる類似性

類似性を捉えるか」そして「いずれの類似性を重視するか」という2つの観点から，直感的に指標を設計することが可能である．

和歌間の類似性指標

図-4をごらんいただきたい．有名な持統天皇の歌と，それを踏まえて詠んだ，藤原定家による本歌取りである．2首を比較すると，持統天皇歌の第3句，第4句に現れた「白妙の衣ほすてふ」は，定家の歌では初句と第2句に現れている．また，持統天皇歌の第2句「夏来にけらし」は定家の歌では少し姿を変えて「夏の来て」となり第3句に現れている．このような類似性を扱うには，句の順序の変化を考慮しなければならない．そこで，一方の歌の句を他方の歌の句と対応付け，対応付けられた5組の句の間で類似度をそれぞれ求め，合計する．句の対応付けは5!＝120通りあるから，その中からこの合計値を最大にする対応付けを選び，その最大値を2つの歌の間の類似度とする．句間の類似度については，上に述べたように，2つの句に共通するパターンを考え，その最大スコアを類似度とする．

ここで(1)パターンの形式と(2)各パターンにスコアを割り当てるパターンスコア関数を定めなければならない．パターンとしては，「なつ★き★」のようなワイルドカード付きパターンを採用した．パターン「なつ★き★」は，文字列「なつのきて」「なつきにけらし」の両方に合致する．これらの共通パターンには，ほかに「★なつ★」「な★き★」「★き★」「★」などがある．一方，パターンスコア関数についてだが，最も単純には，パターン中の★以外の文字数をスコアとするものが考えられる．たとえば，上にあげたパターン「なつ★き★」「★なつ★」「な★き★」「★き★」「★」に対するスコアは，順に，3, 2, 2, 1, 0となる．この関数を採用したとすると，得られる類似度は，2つの文字列間の最長共通部分列の長さに一致する．しかし，文字列の共通の度合いを評価するためには，たとえば，パターン「★なつ★」と「★な★き★」では，文字の連続した「★なつ★」により高いスコアを与えた方がよいと思われる．そこで，★で区切られた塊の個数を数え，その個数に比例したペナルティを科すことにし，「スコア＝文字数－塊の個数×s」とした．ここに，sは$0 < s \leq 1$となるパラメータで，類似歌のサンプルを訓練例として求めた値$s = 0.9$を用いている．

多様な類似性への対応

上で示した類似性指標により多くの類似歌対を拾い出すことができた．だが，類似歌抽出に有効な類似性指標が唯一存在するとは考えにくい．むしろ，研究者の視点に応じて指標を自由に変更し，類似度の高い対をその都度確認する，というシナリオが有効であろう．

実際，この指標では，たとえば，図-5に示す2首の間の類似性をうまく扱えない．この2首の場合，句と句を対応付けてしまったのでは，共通する文字列をうまく拾えないのである．たとえば，「かはづ」を共通文字列として拾うには，『古今集』歌の第1句「かはづなく」と『新古今集』歌の第4句「ゐでのかはづ」を対応付ければよいが，すると「なく」「ゐでの」を拾うことができない．逆に「なく」「ゐでの」を共通文字列として拾うと，今度は「かはづ」「やまぶき」を拾えない．

そこで，句と句を対応付けずに歌全体をながめ，共通して生起する文字列を，順序を問わずに拾い出していくことを考えた．すると，図に示すように「かはづ」「なく」「ゐでの」「やまぶき」「ちりにけり」「はな」といった共通文字列群が浮かび上がってくる．このような文字列のリストを**断片パターン**（fragmentary pattern）と呼ぼう．断片パターンは，形式的には，文字列の多重集合（multiset）として定義される．断片パターンπが文字列wに合致するとは，多重集合π中のすべての文字

嘆きつつ／ひとりぬる夜の／明くるまは／いかにひさしき／ものとかは知る
（『百人一首』53番，右大将道綱母）

秋風に／吹きかへされて／くずの葉の／いかにうらみし／ものとかは知る
（『金葉集』392番，藤原正家）

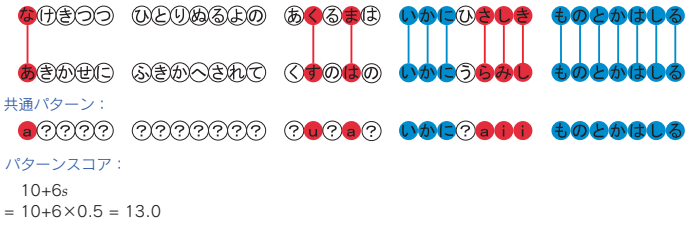

図-6
拡張ハミングパターンでみる類似性

列がオーバーラップなしに文字列w中に現れているときをいう☆3．一方，パターンスコア関数については，比較的単純に，各断片パターンのスコアを，多重集合中の長さ2以上の文字列の長さの総和とした．

もう1つ，図-6の例を見てみよう．ここに示した2首には「いかに★ものとかはしる」という表現が共通している．しかも，その共通の文字列「いかに」「ものとかはしる」は，2首の同じ位置に現れている．この2首の類似性は，31文字の先頭から順に，文字あるいは母音の一致をみていくことでうまくモデル化できる．

下の句を例にとって説明しよう．2つの文字列「いかにひさしきものとかはしる」「いかにうらみしものとかはしる」の文字を左から順に1文字ずつ比較し，文字が一致したときにはその文字を，文字が不一致でも母音が一致したときにはその母音（a, i, u, e, o）を書き，母音も一致しないときには記号？を書くことにすると，「いかに？aiiものとかはしる」を得る．ここで，記号「？」を任意のかな1文字に合致するワイルドカード，母音を表す記号a, i, u, e, oを，それぞれ，その母音を持つ任意のかな1文字に合致するワイルドカードと考えよう．すると「いかに？aiiものとかはしる」は，2つの文字列の共通パターンとみなすことができる．このようなパターンを，拡張ハミングパターンと呼ぶことにする．図-6には，31文字全体で見た際の共通パターンを示している．図では，単純に，パターン中の文字の個数に母音のワイルドカードの個数を$s=0.5$倍して加えたものをパターンのスコアとしている．

このように，パターンの形式を差し替え，適当なパターンスコア関数を与えれば，違った種類の類似性が取り出せる．こうした柔軟性は，本手法の利点である．

☆3 任意に与えられた断片パターンπと文字列wに対して，πがwに合致するかどうかを判定する問題は，NP-完全であることが分かっている（Hori et al. 2001）．

実際，第1筆者のグループでは，拡張ハミングパターンを用いた指標と同様の指標によって，楽曲間の類似性がうまく扱えることを示した（Kadota et al. 2001）．

最適パターン発見としての類似度計算

Σ上の**パターン族**とは，対$<\Pi, L>$をいう．ここに，Πは表現の集合で，各要素を**パターン**と呼び，Lは各パターン$\pi\in\Pi$にその言語$L(\pi)\subseteq\Sigma^*$に対応付ける写像である．文字列wが言語$L(\pi)$に所属するとき，wはパターンπに**合致する**という．Σ上の**SRS**とは，3つ組$<\Pi, L, score>$をいう．ここに，$<\Pi, L>$はパターン族であり，$score$は各パターン$\pi\in\Pi$にそのスコア$score(\pi)\in\mathbf{R}$を対応付ける写像であり，パターンスコア関数と呼ぶ．SRS$<\Pi, L, score>$に関する類似度計算問題を以下のように定義する．

類似度計算問題.
入力：文字列$x, y\in\Sigma^+$.
出力：$\{x, y\}\subseteq L(\pi)$を満たし，$score(\pi)$の値を最大にする$\pi\in\Pi$.

ここでは，入力を2つの文字列x, yとしたが，これを一般化して，「文字列の有限集合$S\subseteq\Sigma^+$に対して，$S\subseteq L(\pi)$を満たし$score(\pi)$を最大にする$\pi\in\Pi$を求めよ」という正例からの最適パターン発見問題を考えることもできる．すなわち，SRSの枠組みは，類似度計算問題と最適パターン発見問題のあいだの架け橋となるものである．

国文学研究上の「発見」

本手法で抽出した類似歌対が契機となって，以下のような「発見」がもたらされた．この成果は，国文学分野の論文にまとめられた．

- 親心の率直な吐露とのみ評価されてきた藤原兼輔の歌（『後撰集』1102番）が，清原深養父（『古今集』585番）

人の親の／心は闇に／あらねども／子を思ふ道に／まどひぬるかな
（『後撰集』1102番，藤原兼輔）

人を思ふ／心は雁に／あらねども／雲居にのみも／なきわたるかな
（『古今集』585番，清原深養父）

図-7

上の歌は，三十六歌仙のひとり，藤原兼輔（877～933）の代表作で，「子を思う親の心情をストレートに表現した，ほとんど無技巧な歌である」という共通理解を得てきた．ところがこの歌，実は，下の清原深養父の歌を踏まえて作られたものと見られるのである．2つの歌を比べてみると，「人…／心は…に／あらねども／…／…るかな」という構造が共通であり，さらに，第2句の「心は…に」の「…」の部分が「やみ／yami／」と「かり／kari／」であり母音がともに［a］［i］となっている点までが共通している．並べて見るとこれだけ似ているにもかかわらずこれまでに指摘がなかったのは，2首に共通するのが自立語というより言い回しの部分であって，研究者の記憶に残りにくいためであろう．このように，兼輔歌は，共通する自立語こそ多くはないものの，深養父歌と，きわめて高い類似性を示している．すると，先の兼輔歌は，単に無技巧といって片づけられないことになる．彼の念頭には，深養父の恋歌があった．兼輔は，そこに詠まれた恋人への一途な愛情を，我が子に向けて「替え歌」に仕立てたのであろう．このような有名な歌でも，付属語や音の共通性を考慮することで，これまで忘れ去られていた一面が発見されることもある．この2首は，『古今集』1,111首と『後撰集』1,425首の間のすべての組合せについて，図-4の指標を用いて類似度を算出した結果，34位に浮かび上がってきたものである．

計算機で抽出した類似歌

の骨組みを利用した，いわば「替え歌」であることを発見した（図-7参照）．これにより，古歌を踏まえた歌作りの一面が明らかになった．

- これまで鎌倉時代中期の成立ではないかと考えられていた『為忠集』と，平安最末期以降の私家集（155集，約10万首）との間で，網羅的に類似歌の抽出を行った結果，室町時代に成立した正徹の『草根集』に，まとまった数の類似歌を見出した．さらに，正徹の弟子である桜井基佐の『基佐集』に『為忠集』に載る歌と同一の歌を見出した．そこで，『為忠集』に現れる人物の考証を併せて行い，『為忠集』の成立が室町時代であることを実証した．

◎差異表現の抽出

2つの和歌の集合に対して，一方には頻出するが，他方には稀であるような文字列を抜き出せば，2つの集合の差異を特徴付ける表現が得られる可能性がある．たとえば，近藤[4]は，『古今集』に詠作者の性別の情報を付加し，性差による表現特徴の抽出を試みている．

この問題は，正例・負例からの最適パターン発見問題として捉えることができる．与えられた文字列の有限集合S, Tと任意のパターン$\pi \in \Pi$に対して，$x = |S \cap L(\pi)|, y = |T \cap L(\pi)|$とおく．すなわち，$x, y$は，それぞれ，集合$S, T$の文字列のうちでパターン$\pi$に合致したものの個数である．この$x, y$を引数にとる関数$f$を仮定し，$f(x, y)$の値が大きいほど良いパターンと考えることにしよう．このfとしては，情報利得，χ^2値，Gini指標などがよく用いられる．

最適パターン発見問題．
入力：文字列の有限集合$S, T \subseteq \Sigma^+$．
出力：$f(x, y)$の値を最大にする$\pi \in \Pi$．ここに，$x = |S \cap L(\pi)|, y = |T \cap L(\pi)|$である．

現実には，最適なパターンだけでなく，$f(x, y)$の値の降順にパターンを整列し，上位のものについて人間が吟味することになる．

ワイルドカード付きパターン族に対する最適パターン発見問題は，一般にNP-困難である．第1筆者らの研究グループでは，DNAの塩基配列やアミノ酸配列の解析を目的に，十分広いクラスのfに対して適用可能な枝刈技法を考案し，このパターン族について最適パターン発見問題を解く実用的手法を開発している（Inenaga et al. 2002）．

一方，ワイルドカード付きパターンを「★w★」のかたちに制限したものを，部分文字列パターンと呼ぶ．ここに，wは長さ1以上の文字列とする．部分文字列パターン族に対する最適パターン発見問題は線形時間で解くことができ，計算量に関する障壁は特にない．そこで，$f(x, y)$の値の高いパターンについて人手で吟味する作業をいかに円滑に進めるかが成功の鍵となる．

nグラム統計を用いた研究

近藤[4]は，『古今集』から詠作者の性差による語彙や表現特徴を抽出する際に，nグラム統計を用いている．ここでのnグラムとは，テキストから切り出した長さnの文字列を指す．このnの値は「2から15まで」などと利用者が指定し，値ごとに，文字列のリストが作成される．たとえば「はるかすみ」という語がテキスト中に生起したとすれば，3グラムのリストには，「はるか」「るかす」「かすみ」が現れる．リストには「るかす」のような無意味な断片も少なからず含まれるため整理が必要であるが，これにはかなりの労力を要する．実際，近藤は，(1) 長さが$n = 3 \sim 7$であり，(2) 2回以上生起し，かつ，(3) 女性歌人による歌での生起頻度が0であるものに制限している．

だが，近藤がnグラム統計を用いたのは，それ以外に文字列分析の手段を持たなかったからに過ぎないのではないか．必要なのは，あくまでテキストの部分文字

列についての統計情報である．筆者らは，nグラムへのこだわりを捨てることによって，この作業の手間を軽減することに成功した．

長さで輪切りにすることなく

テキストの部分文字列の異なり数は，一般にテキスト長の2乗に比例する．nグラム統計では，この大量の部分文字列群を，長さごとに仕分けして，それぞれに統計をとる．このことが，本来見えていたはずの情報を，分断してしまった．

テキストによっては，「はるか」「かすみ」は，すべて「はるかすみ」の一部として生起しているかもしれない．また別のテキストでは，「はるか」「かすみ」は，「はるかすみ」としての生起以外に，「はるかけて」や「あかすみは（明かす身は）」などの一部としての生起を含むかもしれない．文字列とその部分文字列との間には，このような2種類の可能性が存する．ところがnグラム統計では，テキストの部分文字列群を長さで「輪切り」にするために，文字列間のこのような関係が見えてこない．そこで，テキストの部分文字列群を，以下のようにして提示し，利用者の負担を軽減する．

- テキスト部分文字列全体を，ある同値関係に基づいて同値類に分割する．同じ同値類に属する文字列は同じ頻度を持つため，同値類を$f(x, y)$の値の降順に整列しておき，研究者はそれを上位から1つ1つ吟味する（表-1参照）．
- 文字列の左・右に続く文字列を，木構造のかたちで表示させ，容易に辿れるようにする（図-8）．

ここで用いる同値関係の正確な定義は省略するが，あるテキストにおいて「るかす」の生起がすべて「はるかすみ」の一部としての生起であったとすれば，4グラムのリスト中にある「はるかす」「るかすみ」の生起は，すべて「はるかすみ」の一部としての生起である．このとき，これらの部分文字列は，同じ同値類に属する．同値類の個数はテキストの長さに比例する．

上述のアイディアの実現には，SCDAWG（Symmetric Compact Directed Acyclic Word Graph）と呼ばれるデータ構造を用いればよい．SCDAWGは，接尾辞木（suffx tree）などと同様，テキストに対する索引構造の1つで，線形時間・領域で構築可能である．一方，近藤の用いたnグラム統計作成ツールは，NagaoとMori[8]によって開発されたもので，大規模テキストに対しても適用可能なツールとして知られている．文献8）に示されたデータ構造は，Gonnet（1987）やManberとMyers（1990）がそれぞれ独立に考案した接尾辞配列（suffix array）と本質的に同一である．すなわち，テキストに対する接尾辞配列を作成し，それを用いて長さnの部分文字列とその頻度を枚挙する．この接尾辞配列は，接尾辞木と比べ領域が少なくてすむことから，近年注目を集めている．

順位	$f(x, y)$	x	y	同値類
1	-0.9835	27	80	みに
⋮				
11	-0.9851	19	6	はるかすみ るかすみ はるかす るかす
⋮				
187	-0.9863	11	33	つゆの
188	-0.9863	4	0	あやまたれけ やまたれけ またれけ あやまたれ やまたれ
189	-0.9863	4	0	かけか
190	-0.9863	4	0	かしら
⋮				

表-1
『古今集』と『後撰集』の比較（一部）

◎単語分割 vs. 文字列分析

本研究への批判の多くは，意味を扱わず，和歌を単なる文字列として扱っていることに集中する．ほかに文字列分析に徹した研究として，中村[9]や近藤[4]がある．一方，文字列分析によらないアプローチとしては，伊藤[3]や村上ら[7]のように，人手によってテキストを単語に分割し，品詞情報を付与した上で，単語や品詞を単位とした処理を施すものがある．だが，この単語分割は多大な労力を要するだけでなく，専門家であっても非常に困難である．近藤[5]は，この点について，次のように述べている．「古典語にせよ，現代語にせよ，日本語において，一語をどう認定するかは，その基準の立て方にも様々な立場があり，従来から多くの研究がなされてきた．そもそも単位をめぐる基準からして，一通りではない．」

単語や品詞を単位とした処理の是非については，依然として議論が分かれるところであるが，村上と共同研究を行ってきた今西[2]が，次のように記している点，

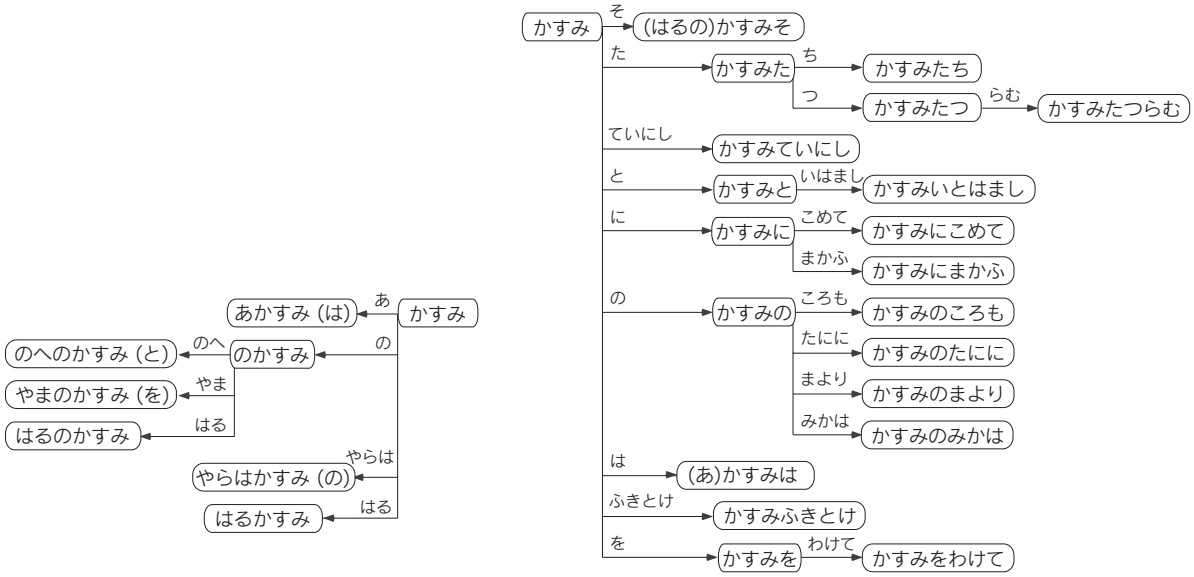

図-8
『古今集』と『後撰集』における「かすみ」に対する左右の文脈木

注目に値しよう．「そもそも品詞という単位，あるいは概念は，日本語にとって必ずしも本来的な要素ではなく，古人の言語意識に沿ったものであるかどうかも保証の限りではない．品詞に代わる，より効果的な尺度はないか．たとえば人物呼称，あるいは品詞の単位を越えた文字列等の探索が必要である．」

◎モビルスーツを装着した上で

国文学研究者の多くは，これまで研究における計算機利用に関してかなり否定的であった．計算機とはいかなるものであり，どう位置付けるべきものなのか，判然としなかったためであろう．しかし，ワープロの普及などを1つの契機として，「単なる道具」としての計算機のイメージが浸透し，少しずつその利用が広がってきた．「コンピュータに文学が分かるものか」といわれて「それでは，あなたの万年筆に文学が分りますか？」とやり返すという有名なエピソードは，そのような流れを端的に表している．

だが「所詮は万年筆」ではいかにも寂しいではないか．本稿では，その思いを込めて「モビルスーツ」と呼んでみた．万年筆からすれば，ずいぶんと出世したものだが，もちろん，道具であることには変わりない．

いくら性能の良いモビルスーツが開発されようと，重要なことは，国文学研究者が，計算機が提示したデータから何を着想するかである．研究者は，常に澄んだ目でデータと向かい合わなければならない．そして，計算機を用いることで用例収集の労力が減った分，データの吟味とその意味付けに時間を費やすべきである．用例の列挙だけでは，決して「研究」とは呼べない．

参考文献
1) 有川節夫ほか: 発見科学の構想と展開, 人工知能学会誌, Vol.15, No.4, pp.595-607 (July 2000).
2) 今西裕一郎: 平安時代物語文の比較計量的研究, 文部科学省科学研究費補助金特定領域研究 (A)「古典学の再構築」第I期公募研究論文集, pp.156-164 (2001).
3) 伊藤鉄也: 源氏物語古写本における異本間の位相に関する研究の展望, 文部省科学研究費補助金特定領域研究 (A)「人文科学とコンピュータ」, 1999年度研究成果報告書 (2000).
4) 近藤みゆき: nグラム統計処理を用いた文字列分析による日本古典文学の研究－『古今和歌集』の「ことば」の型と性差－, 千葉大学文学部『人文研究』, No.29, pp.187-238 (2000).
5) 近藤みゆき: n-gram統計による語形の抽出と複合語－平安時代語の分析から－, 日本語学, Vol.20, No.9, pp.79-89 (Aug. 2001).
6) 村田右富実: 万葉集研究におけるコンピュータ利用の一側面－万葉短歌の字余りを中心に－, 文学・語学, No.171, pp.65-80 (2001).
7) 村上征勝, 今西裕一郎: 源氏物語の助動詞の計量分析, 情報処理学会論文誌, Vol.40, No.3, pp.774-782 (Mar. 1999).
8) Nagao, M. and Mori, S.: A New Method of N-gram Statistics for Large Number of n and Automatic Extraction of Words and Phrases from Large Text Data of Japanese, In Proc. 15th Intern. Conf. on Computational Linguistics (COLING '94), pp.611-615 (1994).
9) 中村康夫: 古典研究のためのデータベース, 国文学研究資料館編原典講読セミナー5, 臨川書店 (2000).
10) Takeda, M., Fukuda, T., Nanri, I., Yamasaki, M. and Tamari, K.: Discovering Instances of Poetic Allusion from Anthologies of Classical Japanese Poems, Theoretical Computer Science, to appear.
11) Takeda, M., Matsumoto, T., Fukuda, T. and Nanri, I.: Discovering Characteristic Expressions in Literary Works, Theoretical Computer Science, to appear.
12) Yamasaki, M., Takeda, M., Fukuda, T. and Nanri, I.: Discovering Characteristic Patterns from Collections of Classical Japanese Poems, New Generation Computing, Vol.18, No.1, pp.61-74 (2000).
（平成14年8月4日受付）